Calculation of Hydration Free Energy for a Solute with Many Atomic Sites Using the RISM Theory: A Robust and Efficient Algorithm

MASAHIRO KINOSHITA,[1] YUKO OKAMOTO,[2] FUMIO HIRATA[2]

[1] *Advanced Energy Utilization Division, Institute of Advanced Energy, Kyoto University, Uji-shi, Kyoto 611, Japan*

[2] *Department of Theoretical Studies, Institute for Molecular Science, Okazaki-shi, Aichi 444, Japan*

Received 8 October 1996; accepted 25 February 1997

ABSTRACT: We have developed an algorithm for solving the reference interaction site model (RISM) equations for water near a solute molecule with many atomic sites (interaction sites). It is a hybrid of the Newton–Raphson and Picard methods and is judiciously constructed. Various considerations are given so that the computer time can be saved as much as possible. The robustness and high efficiency of the algorithm has been demonstrated for calculating hydration free energies of Met-enkephalin (a peptide with 75 sites) with different conformations. The Jacobian matrix is treated as part of the input data, and it has been found that the same matrix can be used for a considerably large set of different conformations of the solute molecule. © 1997 by John Wiley & Sons, Inc. *J Comput Chem* **18:** 1320–1326, 1997

Keywords: algorithm; RISM theory; Jacobian matrix; hydration free energy; peptide

Introduction

A full theory for conformation of biopolymers requires a method for treating the effects of solvent on the induced structures. The reference interaction site model (RISM) theory provides this type of method and allows us to analyze a biopolymer–solvent system on an atomic level. Pettitt and coworkers[1, 2] applied the RISM theory to the calculation of the free energy surface of di- and tripeptides, but they used the superposition approximation in which the entire free energy of a peptide is expressed as a sum of the potential of mean forces between pairs of atoms. The work of Kitao et al.[3] was the first to employ the full RISM theory for free energy analysis of melittin (a small

Correspondence to: M. Kinoshita; e-mail kinoshit@iae,kyoto-u.ac.jp

Contract/grant sponsor: Joint Studies Program of the Institute for Molecular Science

CCC 0192-8651 / 97 / 101320-07

protein) in water. As one would expect, however, a large amount of computational effort was required to solve the basic equations. This problem is a major stumbling block in the elaborate studies based on statistical–mechanical treatment.

Our ultimate goal is to develop a robust and very efficient algorithm for solving the full RISM equations for a biopolymer–aqueous solution system, and the present study is the first step in this direction. We have previously reported an algorithm[4,5] for solving the RISM equations for solvent near a spherical particle to calculate hydration free energy. In this article this algorithm is extended to water near a molecular solute with many atomic sites (interaction sites). The robustness and high efficiency of the extended algorithm is demonstrated for calculating hydration free energies of Met-enkephalin (a peptide with 75 sites) with different conformations.

Theory

In the present article, the subscripts "v" and "s" denote "water" and "solute," respectively. It is assumed that the solute molecules are present at infinite dilution. The calculation process is then split into two steps where bulk water (step 1) and water near a solute molecule (step 2) are treated, respectively. The site–site intermolecular total correlation functions calculated in step 1 are used as input variables for step 2. The calculation in step 1 is performed using the RISM theory improved by Perkyns and Pettitt[6,7] which assures dielectric consistency. We consider step 2 hereafter.

It is assumed that the solute molecule and a water molecule has m and 3 interaction sites, respectively. The site–site Ornstein–Zernike (SSOZ) equation is expressed as:

$$\tilde{\boldsymbol{\eta}}_{sv} = \tilde{\mathbf{w}}_{ss}\tilde{\mathbf{c}}_{sv}\tilde{\mathbf{H}}_{vv} - \tilde{\mathbf{c}}_{sv} \tag{1}$$

$$\tilde{\mathbf{H}}_{vv} = \tilde{\mathbf{w}}_{vv} + \rho_v\tilde{\mathbf{h}}_{vv} \tag{2}$$

where $\tilde{\mathbf{H}}_{vv}$, $\tilde{\boldsymbol{\eta}}_{sv}$, and $\tilde{\mathbf{w}}_{ss}$ are 3×3, $m \times 3$, and $m \times m$ matrices, respectively. $\boldsymbol{\rho}_v$ is the number density matrix of water molecules in the bulk, $\mathbf{h}$ is the matrix of site–site intermolecular total correlation functions, $\mathbf{c}$ is the matrix of site–site intermolecular direct correlation functions, $\mathbf{w}$ is the intramolecular correlation matrix, and "$\sim$" represents Fourier transforms. $\tilde{\mathbf{H}}_{vv}$ depends on properties of the bulk water alone and is part of the input data for step 2. $\tilde{\mathbf{w}}_{ss}$, $\tilde{\mathbf{w}}_{vv}$, $\boldsymbol{\rho}_v$, $\tilde{\mathbf{h}}_{vv}$, and $\tilde{\mathbf{H}}_{vv}$ are symmetrical matrices. The expressions of some of the matrices are:

$$\boldsymbol{\rho}_v = \begin{Vmatrix} \rho_v & 0 & 0 \\ 0 & \rho_v & 0 \\ 0 & 0 & \rho_v \end{Vmatrix} \tag{3}$$

$$\tilde{\mathbf{c}}_{sv} = \begin{Vmatrix} \tilde{c}_{1\mathrm{H}} & \tilde{c}_{1O} & \tilde{c}_{1\mathrm{H}} \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \tilde{c}_{m\mathrm{H}} & \tilde{c}_{mO} & \tilde{c}_{m\mathrm{H}} \end{Vmatrix} \tag{4}$$

$$\tilde{\mathbf{w}}_{ss} = \begin{Vmatrix} 1 & \tilde{w}_{12} & \cdots & \tilde{w}_{1m} \\ \tilde{w}_{21} & 1 & \cdots & \tilde{w}_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{w}_{m1} & \tilde{w}_{m2} & \cdots & 1 \end{Vmatrix} \tag{5a}$$

and:

$$\tilde{w}_{aa'} = \tilde{w}_{a'a} = \sin(kl_{aa'})/(kl_{aa'}) \tag{5b}$$

where k is the wave number and $l_{aa'}$ is the distance between sites a and a' in the solute molecule. For instance, c_{aH} (c_{aO}) is the site–site direct correlation function between site a of the solute molecule and the water hydrogen (oxygen).

The closure equation employed is of the hypernetted-chain (HNC) type given by:

$$c_{ab}(r) = \exp\{-u_{ab}(r)/(k_BT) + \eta_{ab}(r)\} - \eta_{ab}(r) - 1 \quad a = 1, \ldots, m;\ b = \mathrm{H}, O \tag{6}$$

where $u_{a\mathrm{H}}$ (u_{aO}) is the pair potential between site a of the solute molecule and the water hydrogen (oxygen), and k_B is the Boltzmann constant.

The solvation free energy for the solute molecule, $\Delta\mu_s$, is calculated from[3,8,9]:

$$\Delta\mu_s/(k_BT) = 4\pi\int_0^\infty F(r)\,dr \tag{7a}$$

$$F(r) = \sum_{a=1}^{m}\sum_{b=\mathrm{H},O}\rho_b r^2\left[\{h_{ab}(r)\}^2/2 - c_{ab}(r) - h_{ab}(r)c_{ab}(r)/2\right] \tag{7b}$$

$$\rho_{\mathrm{H}} = 2\rho_v,\ \rho_O = \rho_v \tag{7c}$$

The site–site correlation functions $h_{ab}(r)$ and $c_{ab}(r)$ are calculated by solving eqs. (1) and (6).

Algorithm

ITERATION STRATEGY

A sufficiently long range, r_L, is divided into N mesh points ($r_i = i\delta r, i = 0, 1, \ldots, N-1$; $\delta r = r_L/N$) and all the functions are represented by their values on these points. In the present analysis, δr and N are set at $0.02d$ ($d = 0.28$ nm) and 512, respectively. We iterate on $\eta_{ab}(r_i)$ (there are $2m$ distinct pairs). $\eta_{ab}(r_i)$ for $r_i \leqq D$ ($D << r_L$) are decomposed into ''coarse'' and ''fine'' variables using the projective representation described in our previous articles.[4,10] $\eta_{ab}(r_i)$ for $r_i > D$ are treated as fine variables. The coarse and fine variables are converged in the inner Newton–Raphson and outer Picard loops, respectively. The values of i_{Bt} and κ_t ($t = 1, \ldots, 2m$; $2i_{Bt}\delta r$ and κ_t denote the width and the number of the root basis functions, respectively) can all be different.[10] We note that the dimensionality of the Jacobian matrix can be greatly reduced by choosing a large value for i_{Bt}. D is chosen such that:

$$c_{ab}(D) \sim -\eta_{ab}(D) - 1 \tag{8a}$$

$$\partial c_{ab}(r_i)/\partial \eta_{ab}(r_i) \sim -1 \quad \text{for } r_i \leqq D \tag{8b}$$

The Jacobian matrix is then determined from bulk properties and the intramolecular correlations of the solute molecule alone and independent of the solute molecule–water correlations.[4,10] It is treated as part of the input data (it is determined in the calculation of step 1).

As given by eq. (4), the first and third columns of $\tilde{\mathbf{c}}_{sv}$ are identical, and this is also true for $\tilde{\boldsymbol{\eta}}_{sv}$. Using this feature of the matrices, we rewrite the SSOZ equation as follows so that it can be handled most efficiently:

$$(\tilde{\eta}_{sv})ij = \left(\tilde{\mathbf{w}}_{ss}\,\tilde{\mathbf{c}}_{sv}\,\tilde{\mathbf{H}}_{vv}\right)ij - (\tilde{\mathbf{c}}_{sv})ij \tag{9}$$

$$\left(\tilde{\mathbf{c}}_{sv}\tilde{\mathbf{H}}_{vv}\right)nj = \xi_{\mathrm{H}}(j)(\tilde{\mathbf{c}}_{sv})n1 + \xi_O(j)(\tilde{\mathbf{c}}_{sv})n2 \qquad n = 1, \ldots, m;\ j = 1, 2 \tag{10}$$

$$\left(\tilde{\mathbf{w}}_{ss}\tilde{\mathbf{c}}_{sv}\tilde{\mathbf{H}}_{vv}\right)ij = \sum_{n=1}^{m} (\tilde{\mathbf{w}}_{ss})in\left(\tilde{\mathbf{C}}_{sv}\tilde{\mathbf{H}}_{vv}\right)nj \qquad i = 1, \ldots, m;\ j = 1, 2 \tag{11}$$

$$\xi_{\mathrm{H}}(1) = \left(\tilde{\mathbf{H}}_{vv}\right)11 + \left(\tilde{\mathbf{H}}_{vv}\right)13 \tag{12a}$$

$$\xi_{\mathrm{H}}(2) = \left(\tilde{\mathbf{H}}_{vv}\right)12 + \left(\tilde{\mathbf{H}}_{vv}\right)32 \tag{12b}$$

$$\xi_O(1) = \left(\tilde{\mathbf{H}}_{vv}\right)21 \tag{12c}$$

$$\xi_O(2) = \left(\tilde{\mathbf{H}}_{vv}\right)22 \tag{12d}$$

where $(\mathbf{X})ij$ represents the (i, j)-element of the matrix $\mathbf{X}$, and $\xi_{\mathrm{H}}(j)$ and $\xi_O(j)$ are read from a data file.

Our next concern is to derive an analytical expression of the Jacobian matrix. It can be shown from eqs. (9)–(12) that:

$$\partial(\tilde{\boldsymbol{\eta}}_{sv})ij/\partial(\tilde{\mathbf{c}}_{sv})n1 = \xi_{\mathrm{H}}(j)(\tilde{\mathbf{w}}_{ss})in - \delta_{in}\delta_{j1}, \qquad i, n = 1, \ldots, m;\ j = 1, 2 \tag{13a}$$

$$\partial(\tilde{\eta}_{sv})ij/\partial(\tilde{\mathbf{c}}_{sv})n2 = \xi_O(j)(\tilde{\mathbf{w}}_{ss})in - \delta_{in}\delta_{j2} \tag{13b}$$

where δ denotes Kronecker's delta. We note that the right-hand sides of eq. (13) are completely independent of the solute molecule–water correlations. An analytical expression of the matrix is then derived and arranged in compact form so that the matrix can be constructed quite efficiently using the Fast Fourier Transform[11] (FFT).

At each Newton–Raphson iterative step, the linear set of equations written as

$$\mathbf{Jx} = \mathbf{b} \tag{14}$$

must be solved for $\mathbf{x}$ ($\mathbf{J}$ is the Jacobian matrix, $\mathbf{x}$ is the column vector the components of which are unknown corrections of the independent variables, and $\mathbf{b}$ is the column vector comprising known numbers). This is solved using Crout's method for the LU (L and U represent ''lower triangular'' and ''upper triangular,'' respectively) decomposition followed by the forward and back substitutions.[11] The linear set of equations is subsequently solved with the same $\mathbf{J}$ but a different right-hand side $\mathbf{b}$ for the Newton–Raphson iterations. This can be done efficiently without calculating the inverse of $\mathbf{J}$.[11] Furthermore, because $\mathbf{J}$ is diagonally dominant (this is also a great advantage of our algorithm) in our case, ''pivoting''[11] is not required at all.

Care must be taken in handling the long-range Coulombic potentials. We define the short-range parts of the functions by [12,13]:

$$c_{ab}^{SR}(r) = c_{ab}(r) + f_{ab}(r),\ a = 1, \ldots, m,\ b = \mathrm{H}, O \tag{15a}$$

$$\eta_{ab}^{SR}(r) = \eta_{ab}(r) - f_{ab}(r) \tag{15b}$$

$$f_{ab}(r) = q_a q_b \mathrm{erf}(\alpha r)/(k_B T r) \tag{15c}$$

where q_a and q_b are partial charges on sites a and b, respectively, $\mathrm{erf}(x)$ is the error function, and α is a constant in the range from 1 to 2. The FT of Eq.

(15) yields:

$$\tilde{c}_{ab}^{SR}(k) = \tilde{c}_{ab}(k) + \tilde{f}_{ab}(k) \quad (16a)$$

$$\tilde{\eta}_{ab}^{SR}(k) = \tilde{\eta}_{ab}(k) - \tilde{f}_{ab}(k) \quad (16b)$$

$$\tilde{f}_{ab}(k) = 4\pi q_a q_b \exp\{-k^2/(4\alpha^2)\}/(k_B T k^2) \quad (16c)$$

We note that $\tilde{f}_{ab}(k)$ diverges as $k \to 0$. The numerical forward and back transforms are applied to $c_{ab}^{SR}(r)$ and $\tilde{\eta}_{ab}^{SR}(k)$, respectively, and the divergent term is handled analytically. The great advantage of the use of eqs. (15c) and (16c) is that both $c_{ab}^{SR}(r)$ and $\tilde{\eta}_{ab}^{SR}(k)$ decay very rapidly (exponentially) as $r \to \infty$ and $k \to \infty$, so they can be accurately transformed numerically with a sufficiently small integration range. We have tested another set of $f_{ab}(r)$ and $\tilde{f}_{ab}(k)$[4,5]:

$$f_{ab}(r) = \{1 - \exp(-r)\} q_a q_b/(k_B T r) \quad (15d)$$

and:

$$\tilde{f}_{ab}(k) = 4\pi q_a q_b/\left\{k^2(k^2+1)k_B T\right\} \quad (16d)$$

and found that r_L with $\delta r = 0.02d$ ($d = 0.28$ nm) and $N = 512$ is too short: the charge neutrality is poorly satisfied, the long-range behavior of $F(r)$ [see eq. (7)] becomes pathological, and the hydration free energy is quite erroneous ($N = 1024$ or $N = 2048$ is required). The renormalization technique[7,13] suffers from a similar shortcoming.

INITIAL GUESS

There is no need to worry much about the initial guess for the iteration variables. We propose the simple setting expressed as

$$\eta_{ab}(r) = q_a q_b/(k_B T r) \quad \text{for } r \geqq r_{ab} \quad (17a)$$

$$\eta_{ab}(r) = q_a q_b/(k_B T r_{ab}) \quad \text{for } r \leqq r_{ab} \quad (17b)$$

If q_a is zero, this setting leads to the ideal gas condition. No careful choice for r_{ab} is required. We simply propose:

$$r_{ab} \sim d \quad \text{for } q_a q_b > 0 \quad (18a)$$

$$r_{ab} \sim r_0 \quad \text{for } q_a q_b < 0 \quad (18b)$$

where:

$$u_{ab}(r_0) \sim 0 \quad (18c)$$

The parameters, I_{Bt} and κ_t, are chosen such that $i_{Bt}\kappa_t \delta r \sim r_{ab}$ (i_{Bt} is set to 4). Optimization of r_{ab}, i_{Bt}, and κ_t will lead to more robustness and higher efficiency, but we have not optimized them because the calculation is already sufficiently stable and efficient.

Numerical Test

A computer program was written in FORTRAN statements to demonstrate the robustness and high efficiency of the algorithm. One of the simplest peptides, Met-enkephalin {Tyr−Gly−Gly−Phe−Met}, previously studied as a benchmark for the protein folding problem,[14−16] is chosen for the demonstration. All calculations are performed on an interactive workstation at the Institute for Molecular Science (IBM RS6000/3CT: 64 MB). The potential energy functions and parameters employed for Met-enkephalin are described in Ref. 16. The model for the water molecule was the extended simple point charge (SPC/E) model.[17] The temperature was set at 298.15 K. $u_{ab}(r)$ has the form:

$$u_{ab}(r) = q_a q_b/r + 4\varepsilon_{ab}\left\{(\sigma_{ab}/r)^{12} - (\sigma_{ab}/r)^6\right\}$$
$$a = 1, \ldots, m;\ b = \mathrm{H}, O \quad (19)$$

and the standard combination rule:

$$\varepsilon_{ab} = (\varepsilon_a \varepsilon_b)^{1/2}, \quad \sigma_{ab} = (\sigma_a + \sigma_b)/2 \quad (20)$$

is employed for calculating the Lennard−Jones potential parameters. Three different conformations of Met-enkephalin are tested and shown in Figure 1. Conformation A is the lowest energy conformation in gas phase determined previously.[15,16] Conformations B and C are two examples of more fully extended ones. Conformational energies for conformations A, B, and C are −12.0 kcal/mol,[16] −2.8 kcal/mol, and 0.8 kcal/mol, respectively. In this article we are more interested in convergence properties of the algorithm than in physicochemical implications of the results calculated. Physicochemical discussions on the results will be published elsewhere.[18]

The convergence criterion is set so that the hydration free energy can be calculated with the accuracy ±0.3 kcal/mol (0.6 kcal/mol ~ $k_B T$). We note that this is a very severe criterion. Nevertheless, with the crude initial guess described above, convergence is achieved in 80, 80, and 84 total outer-loop iterations for conformations A, B, and C, respectively. When the converged solution for confomation B (C) is used as the initial guess in the calculation for conformation C (B), 61 (57) outer-loop iterations are required to achieve convergence. The construction and the *LU* decomposi-

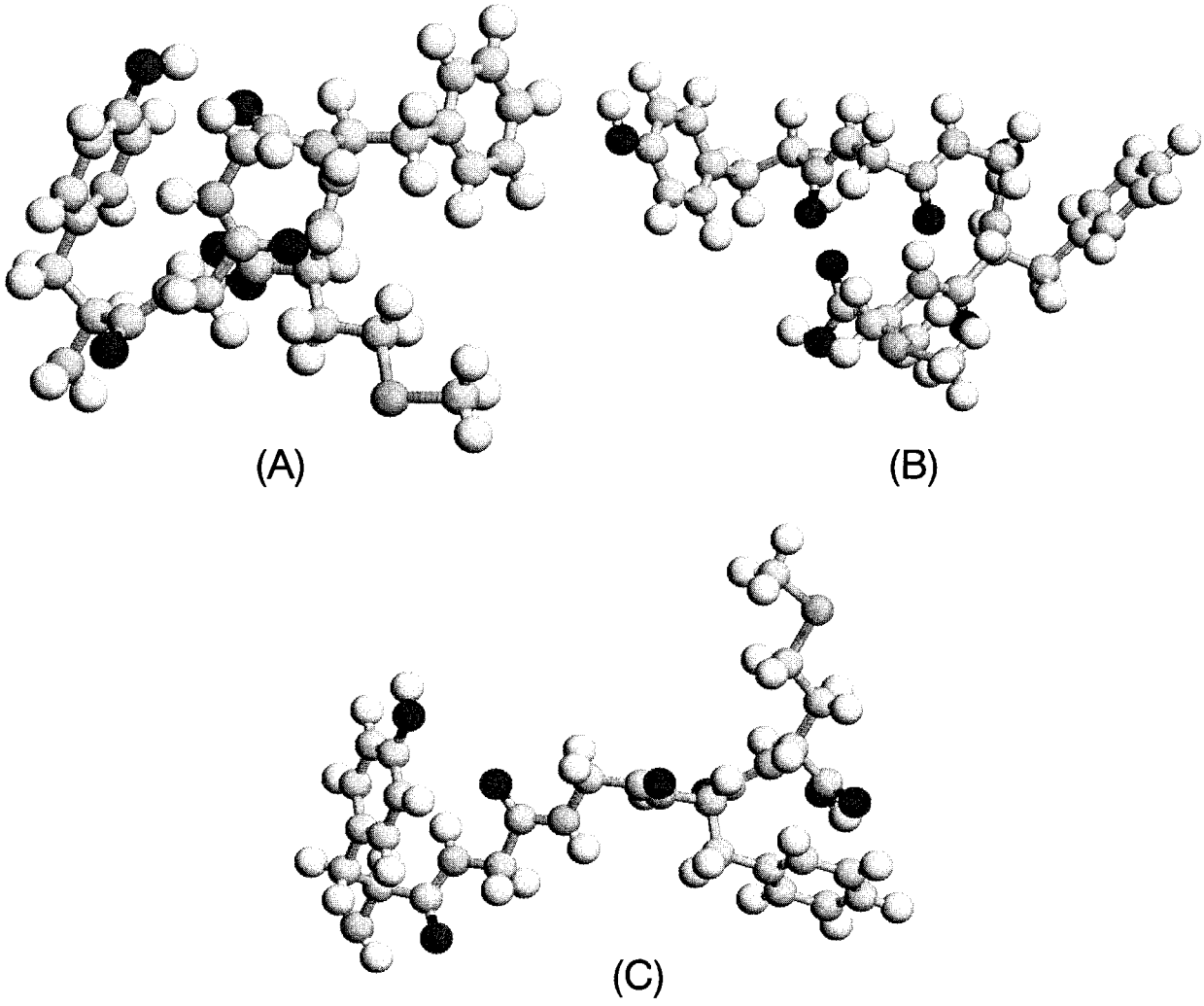

FIGURE 1. Conformations of Met-enkephalin tested (black spheres represent oxygen atoms). (A) Conformation A. (B) Conformation B. (C) Conformation C.

tion of the Jacobian matrix (the dimensionality of the matrix is 1340 for $i_{Bt} = 4$) are finished in ~ 2.3 min and ~ 4.7 min, respectively. The computer time required per 20 outer-loop iterations is ~ 1.1 min. We note that these are not for the central processing unit (CPU) time but for the time which has elapsed. Thus, the calculation of the hydration free energy for Met-enkephalin by solving the full RISM equations can be completed with minor computational effort. (When i_{Bt} is set to 6, the dimensionality of the Jacobian matrix is 857. For conformation A, for example, convergence is achieved in 108 outer-loop iterations. The construction and the LU decomposition of the matrix are finished in ~ 2.0 min and ~ 0.5 min, respectively. The computer time required per 20 outer-loop iterations is ~ 0.9 min.)

The site–site radial distribution functions which give useful information on the water structure near Met-enkephalin are also calculated, but they are discussed in a separate article.[18] Hydration free energies for conformations A, B, and C are 196.8 kcal/mol, 185.4 kcal/mol, and 176.8 kcal/mol, respectively. Considering the total energy (i.e., ''conformational energy'' + ''hydration free energy''), the extended conformations (conformations B and C) are more stable in water, which is in qualitative accord with the experimental observations[18,19] (data obtained by nuclear magnetic resonance [NMR] imaging).

The most time-consuming part of the algorithm is the LU decomposition of the Jacobian matrix. The matrix is dependent on the conformation of the solute molecule. However, we have found that

the same matrix can be used (i.e., the construction and the LU decomposition can be skipped) even for the next calculation, unless the conformation has been considerably changed. For example, the matrix constructed and LU decomposed for one of the three conformations (A, B, and C) can be used in calculations for the other two conformations with a simple technique expressed as:

$$x_I^{\mathrm{NEW}} = x_I^{\mathrm{OLD}} + \zeta \Delta x_I, \quad 0 < \zeta < 1 \qquad (21)$$

where x_I (for $i_{Bt} = 4$, $I = 1, \ldots, 1340$) are the independent variables for the Newton–Raphson iterations, Δx_I are the corrections [the components of the column vector $\mathbf{X}$ in eq. (14)], the superscripts ''NEW'' and ''OLD'' represent the new and old values, respectively, and we have found that 0.75 is small enough for ζ. The convergence properties are not deteriorated by this treatment (only a few tens of inner-loop iterations are additionally required), which leads to considerable saving of computer time. We have tested several other conformations of Met-enkephalin and found that the same matrix can be used for a considerably large set of different conformations.

The conventional algorithm is extremely sensitive to the initial guess chosen.[3] Hence, convergence under a specified condition must be achieved in a far more arduous manner. In the first step, the equations are solved with no intramolecular constraints and atomic partial charges. In the next step, the constraints that are weighted by some factors in the range from 0 to 1 are imposed on the distance between atomic pairs, and the factors are gradually increased to 1. In the final step, the atomic charges are added to the sites step-by-step with a small increment of $\sim 5\%$. Hundreds of iterations are required to obtain convergence in each new calculation using the last solution as the initial guess. In contrast, our algorithm allows us to use the full constraints and atomic charges in the first calculation: starting from a crude initial guess leads to convergence in less than 10^2 iterations. We have found that our algorithm is a few orders of magnitude faster.

Conclusion

We have developed a robust and very efficient algorithm for solving the full RISM equations for water near a molecular solute with many atomic sites (interaction sites). It is a hybrid of the Newton–Raphson and Picard methods. The Jacobian matrix is treated as part of the input data. It is constant and diagonally dominant, and eq. (14) is efficiently solved for the inner-loop (Newton–Raphson) iterations. The same matrix can be used for a considerably large set of different conformations of the solute molecule. The long-range Coulomb potentials are handled in a special manner so that the number of mesh points N can be minimized. Sufficient stability is assured even with the crude initial guess expressed as eqs. (17) and (18). Because the Jacobian matrix is part of the input data, it is completely independent of the initial guess. The hybrid algorithm is now capable of treating a molecular solute with many interaction sites (e.g., a peptide) with minor computational effort on an interactive workstation.

We intend to consider the following developments: to combine the hybrid algorithm with a powerful conformational sampling method[15,16] (to find the lowest energy conformation of a peptide in water); to extend the algorithm to a water–protein system; and to include ions in water. Even for a small protein $m \sim 10^3$ and a straightforward extension of the algorithm gives rise to a very large Jacobian matrix. We believe that even this problem can be overcome by judicious construction of a hybrid algorithm (there is no need to apply the Newton–Raphson method to all site–site correlation functions). We are now making further progress toward achieving these developments.

Acknowledgment

Calculations from this study were performed using the computational facilities at IMS.

References

1. B. M. Pettitt and M. Karplus, *Chem. Phys. Lett.*, **121**, 194 (1985).
2. G. L. Ramé, W. F. Lau, and B. M. Pettitt, *Int. J. Peptide Prot. Res.*, **35**, 315 (1990).
3. A. Kitao, F. Hirata, and N. Gō, *J. Phys. Chem.*, **97** 10233 (1993).
4. M. Kinoshita and F. Hirata, *J. Chem. Phys.*, **104**, 8807 (1996).
5. M. Kinoshita and F. Hirata, *J. Chem. Phys.* **106,** 5202 (1997).
6. J. Perkyns and B. M. Pettitt, *Chem. Phys. Lett.*, **190**, 626 (1992).
7. J. Perkyns and B. M. Pettitt, *J. Chem. Phys.*, **97**, 7656 (1992).
8. H. A. Yu and M. Karplus, *J. Chem. Phys.*, **89**, 2366 (1988).
9. H. A. Yu, B. Roux, and M. Karplus, *J. Chem. Phys.*, **92**, 5020 (1990).

10. M. Kinoshita and D. R. Bérard, *J. Comput. Phys.*, **124**, 230 (1996).
11. W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes, 2nd Ed.*, Cambridge University Press, Cambridge, UK, 1992, pp. 34–40.
12. K. C. Ng, *J. Chem. Phys.*, **61**, 2680 (1974).
13. G. P. Morriss and D. MacGowan, *Molec. Phys.*, **58**, 745 (1986).
14. Z. Li and H. A. Scheraga, *Proc. Natl. Acad. Sci. USA*, **84**, 6611 (1987).
15. Y. Okamoto, T. Kikuchi, and H. Kawai, *Chem. Lett.*, 1275 (1992).
16. U. H. E. Hansmann and Y. Okamoto, *J. Comput. Chem.*, **14**, 1333 (1993).
17. H. J. C. Berendsen, J. R. Grigera, and T. P. Straatsma, *J. Phys. Chem.*, **91**, 6269 (1987).
18. M. Kinoshita, Y. Okamoto, and F. Hirata, *J. Chem. Phys.* (in press).
19. W. H. Graham, E. S. Carter II, and R. P. Hicks, *Biopolymers*, **32**, 1755 (1992).